UNCLASSIFIED

AD-A236 015

AR-006-420

(2)

# Information Technology Division

DTIC

S JUN 0 4 1991 D

C

RESEARCH REPORT
ERL-0520-RR

## n REASONS WHY PRODUCTION-RULES ARE INSUFFICIENT MODELS FOR EXPERT SYSTEM KNOWLEDGE REPRESENTATION SCHEMES

by

S. J. Cosgrove

APPROVED FOR PUBLIC RELEASE

ELECTRONICS RESEARCH LABORATORY

91-01064

UNCLASSIFIED

UNCLASSIFIED

AR-006-420

DSTO AUSTRALIA

# Electronics Research Laboratory

## Information Technology Division

RESEARCH REPORT
ERL-0520-RR

### n REASONS WHY PRODUCTION-RULES ARE INSUFFICIENT MODELS FOR EXPERT SYSTEM KNOWLEDGE REPRESENTATION SCHEMES

by

S. J. Cosgrove

ABSTRACT (U)

The majority of expert system shells and fully developed expert system packages are founded upon simple observation and hypothesis production-rules. This paper investigates the use of such knowledge representation schemes by analysis of a simple, incomplete plant classification expert system. Furthermore, a new hierarchical knowledge representation scheme known as the n-Cube, which is suitable for classification applications, is discussed with reference to the disadvantages of rule-based systems.



Feb 91 COPY NO 5

APPROVED FOR PUBLIC RELEASE

POSTAL ADDRESS: Director, Electronics Research Laboratory, PO Box 1600, Salisbury, South Australia, 5108. ERL-0520-RR

UNCLASSIFIED

91 3 015

## CONTENTS

# 1 INTRODUCTION

The first reported expert systems had production-rule knowledge representation schemes, examples of which are Mycin [Shortliffe76], Prospector [Duda79] and R1 [McDermot82]. Production-rules are conceptual models of human reasoning which have many undesirable features. Even so, they are still widely used by expert system developers and form the basis of the majority of commercial expert system shell building kits. The major reasons for this loyalty to unstructured production-rules are as follows:-

1) Little effort is required in understanding their basic concepts.

2) They are well understood by knowledge engineers.

3) It is easy for experts to reason in a form similar to production-rules, therefore it is thought that knowledge acquisition is made simple.

4) Rapid prototyping is achievable using simple rule-based systems.

5) The addition of "new" knowledge and the removal of "old" knowledge can be straightforward using a rule-based expert system (this is not necessarily the case).

6) Historically, expert systems have been developed by these conceptual models.

Although rule-based expert systems may seem, at first glance, to be the ideal approach, they are far from satisfactory. As a result, modelling techniques such as Semantic Networks [Findler79], Predicate Calculus [Bundy83], Frames [Winston84] and more recently n-Cube [Cosgrove90] have been developed. In section 2, a simple outline of production-rule representation schemes is given. Section 3 then discusses the limitations of such schemas by analysis of a simple production-rule knowledge-base. Section 4 describes the n-Cube knowledge representation scheme, after which section 5 shows how the the n-Cube's conceptual model and search mechanisms can eliminate the disadvantages associated with production-rule schemas.

## 2 RULE-BASED KNOWLEDGE REPRESENTATION SCHEMES

Production-rules are a model of a domain expert's reasoning process. A simple example of such a rule is as follows:-

IF X1 AND X2 THEN Y.

Here X1 and X2 are the antecedents (known facts or knowledge that have to be validated) and Y a suggestion, hypothesis, consequence, action or conclusion. The relationships between a number of production-rules are usually understood by analysis of AND/OR trees [Winston79], a simple example of which is shown in figure 1. The semantics of this tree are as follows:-

| | |
|---|---|
| c is true if and only if: | f and g are true. |
| a is true if and only if: | b and c are true |
| or | d is true |
| or | e is true. |

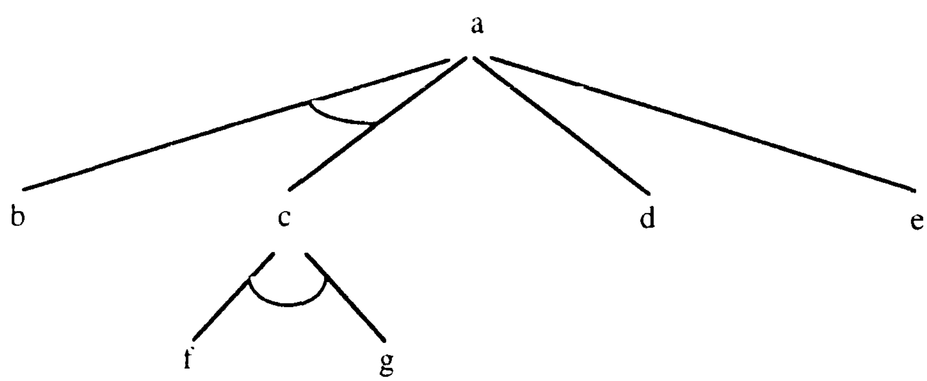


Figure 1. A simple AND/OR graph

This graphical description of the rule structure assists in knowledge-base analysis and design. For instance, high quality graphical tree representations are generated in commercial expert system shell development tools such as NEXPERT [Neuron88]. This graphical approach simplifies knowledge-base development, unfortunately, it is impractical for complex interrelated rule structures consisting of either basic propositional logic or hybrid rule/fact schemas.

## 2.1 Propositional Logic

The simplest form of production-rules are based upon propositional logic [Luger89], for example the rules which model the propositional sequence of months of the year are as follows:-

If the current month is January Then the next month is February.

If the current month is February Then the next month is March.

etc. ( 12 rules in total).

This static schema is inflexible therefore knowledge-base restructuring can be time consuming. In addition, propositional logic has expensive memory storage requirements which may lead to poor system performance.

## 2.2 Hybrid Rule/Fact Schemas

Hybrid rule/fact relationships (also known as Predicate Calculus ) have been developed [Bundy83] to overcome the disadvantages of rule-based propositional schemas. Hence, the above propositional knowledge-base can be modelled as follows:-

If the current month is MONTH1

And MONTH2 follows MONTH1

Then the next month is MONTH2.

(1 rule).

February follows January.

March follows February.

etc. (12 facts in total).

This schema is more versatile than propositional logic, even so it has a number of disadvantages in common with propositional logic rule-based systems. These undesirable features are discussed in section 3.

# 3 THE LIMITATIONS OF RULE BASED KNOWLEDGE REPRESENTATION SCHEMES

Although rule-based expert systems can be created rapidly, they have a number of undesirable features. To highlight these a simple and incomplete expert system, shown in figure 2, will be analysed. This has been used as a tutorial [Minasi90a, Minasi90b] but it does not investigate any of the downfalls associated with such schemas. The tutorial rule-based expert system was designed by an experienced knowledge engineer; even so it has many undesirable features. Hence, if a novice attempted to construct such a rule-based system, then there is a high probability that the performance of the completed system would be far from satisfactory.

| | | |
|---|---|---|
| R1. | If | has green leaves |
| | and | has yellow flowers |
| | and | is a herb |
| | then | is a daffodil. |
| R2. | If | has green leaves |
| | and | has no flowers |
| | and | is a tree |
| | and | is evergreen |
| | then | is a douglas fir. |
| R3. | If | is a mushroom |
| | then | is a jack o'lantern. |
| R4. | If | has green leaves |
| | and | has no flowers |
| | and | is seaweed |
| | then | is a sea lettuce. |
| R5. | If | has green leaves |
| | and | has no flowers |
| | and | is a herb |
| | and | is land based |
| | then | is a lettuce. |
| R6. | If | has green leaves |
| | and | is a tree |
| | and | is land based |
| | and | is deciduous |
| | then | is a pin oak. |
| R7. | If | height is greater than 5 feet |
| | and | stem is woody |
| | then | is a tree. |
| R8. | If | stem is not woody |
| | then | is a herb. |
| R9. | If | is a tree |
| | and | has needles |
| | then | is evergreen |
| R10. | If | is a tree |
| | and | has green leaves |
| | then | is deciduous. |
| R11. | If | produces spores |
| | and | is not green |
| | and | is land based |
| | then | is a mushroom. |
| R12. | If | is not land based |
| | | then is a seaweed |

Figure 2. A Simple Plant Identifier Rule-Base

### 3.1 Failure due to Inconsistency

Analysis of the tutorial expert system, shown in figure 2, identifies that rule R2 and rule R9 contradict each other. For example, rule R2 can only succeed if "is evergreen" is true. Furthermore, "is evergreen" is true if and only if rule R9 is true. This requires "has needles" to be true. Hence, rule R2 can only succeed if and only if "has needles" and "has green leaves" are true. This can never occur, consequently rule R2 can never succeed.

The unstructured nature of simple rule-based schemas facilitates ad-hoc addition and deletion of rules, addition and deletion of antecedents, and modifications to hypotheses. As a result, the probability of failure due to inconsistency increases with rule complexity and knowledge-base size.

### 3.2 Duplication

A second undesirable feature of rule-based expert systems is duplication. For example, the rules that describe the plant "douglas fir"can be displayed in the form of an AND graph as shown in figure 3. This graph shows that "douglas fir" is true if all the children in the tree are true. By inspection , the fact "is a tree" occurs twice which indicates that there is duplication in the rule structure. In addition, the mutually exclusive pair ("has green leaves" and "has needles") emphasise the inconsistency discussed above.

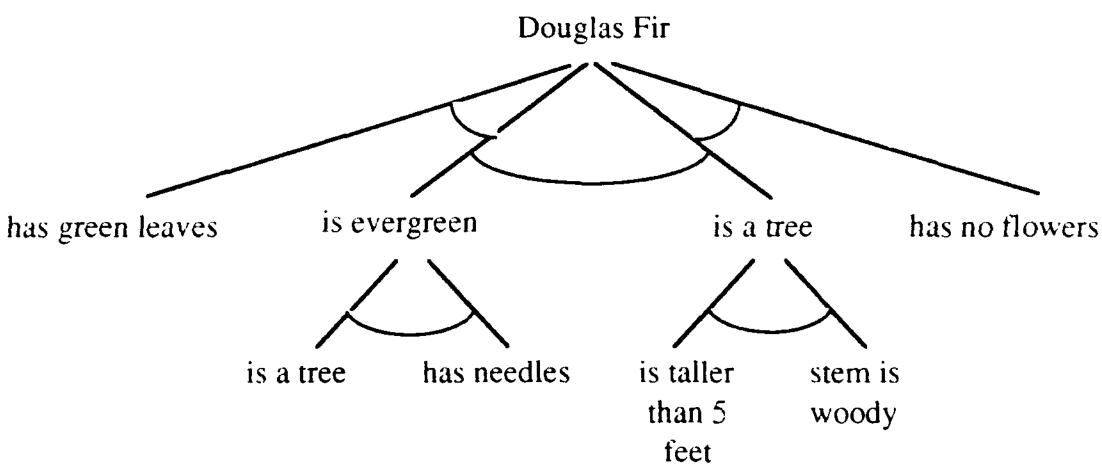


Figure 3. AND Graph of the rules associated with Douglas Fir

Duplication leads to vague explanations of conclusions. Furthermore, it causes unnecessary questioning which in turn can lead to the searching of irrelevant rules. Consequently, this undesirable rule-based schema feature affects both system performance and user confidence.

### 3.3 Incorrect Results

Another disadvantage of simple rule-based expert systems is that they may suggest incorrect results or search along incorrect paths. In so doing, silly or irrelevant questions may be asked. For example, if the plant to be identified was a sea lettuce and rule R8 was investigated before rule R4, then plant type herb would become true and the system could investigate sub classes of herb. This is obviously the wrong search space to focus upon, but because the fact "is land based" has been overlooked in rule R8 the system deduces that the plan is a herb. Hence, the rule firing and rule interactions must be analysed to ensure that the system's performance is not susceptible to incorrect results. Unfortunately this is an iterative and time consuming process which in turn could lead to further undesirable features.

### 3.4 Updating Difficulties

The final drawback of simple rule-based expert systems is due to the difficulties in upgrading or adding new knowledge. For instance, if the "is a mint" rule (shown below) was added to the knowledge-base, then a detailed analysis of the existing rules must be undertaken so that conflicts and inconsistencies do not occur. If this is not accomplished, subtle errors that are difficult to detect and resolve may arise and therefore drastically reduce the system's performance.

| | | |
|---|---|---|
| R 13. | If | is a herb |
| | and | has no flowers |
| | and | has has small leaves |
| | then | is a mint. |

The problem with adding a rule, such as rule R13, is that a conflict arises between this new rule and rule R4. This is due to the knowledge engineer overlooking the fact that "is land based" does not occur in the rule which defines herb. Even if this was not overlooked then rule R5 would have to be revised for consistency. This revised rule requires at least one additional antecedent as shown by the following rule:-

| | | |
|---|---|---|
| R5. | If | has green leaves |
| | and | does not have small leaves |
| | and | has no flowers |
| | and | is a herb |
| | and | is land based |
| | then | is a lettuce. |

# 4 THE n-CUBE KNOWLEDGE REPRESENTATION SCHEME

To overcome the difficulties associated with simple production-rule knowledge-based systems, a number of structured, flexible schemas have been devised. This section discusses the n-Cube knowledge representation scheme and shows how it can overcome the disadvantages of production-rule systems applied to classification problems.

The n-Cube architecture is basically a classification decision tree consisting of a Parent Class (P) which is the root of the tree, a set of Sub-Classes (S) which are the tree nodes, a set of Hypotheses (H) which are the leaves of the tree, an associated Rule Set (R) that locally identifies the conditions required to validate either P or S or H, additional Inheritance (I) default information defined by True (T) statements and False (F) statements. The structure of this hierarchical architecture, shown in figure 4 , is defined below.

The n-Cube's root node, in which $R \Rightarrow P \Rightarrow I$, is known as level 1 The n-Cube's leaf nodes in which $R \Rightarrow H \Rightarrow I$ is known as level m. Furthermore, assume every branch in the n-Cube has an equal depth of m. Thus, each node in the n-Cube's structure is represented by a quadruple of the form :-

$$(\alpha, R, T, F) \text{ where } \alpha \text{ can be either: P, S or H.}$$

Then by induction, the structure can be defined as follows:-

Level 1 is the set $\{(P, \phi, T, \phi)\}$ containing a single quadruple, where $\phi$ is the empty set.

Level n is the set $\{(Sn,i, Rn,i, Tn,i, Fn,i)\}$ $1 < n < m$, $i = 0 ..... k$, where for each i there is some j such that $Sn,i = \{S \in Sn-1,j \,;\, Rn,i\}$ .

Level m is the set $\{(Hm,i, Rm,i, Tm,i, Fm,i)\}$ $i = 0 ..... k$, where for each i there is some j such that $Hm,i = \{S \in Sm-1,j \,;\, Rm,i\}$ .

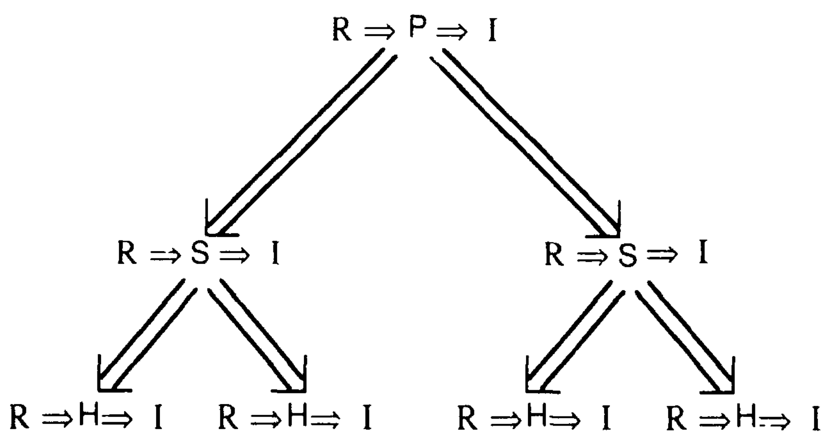


Figure 4. The structure of the n-Cube knowledge representation scheme

The n-Cube syntax used to model the above structure is based upon a 3-tuple as follows:-

Level 1 consists of: rule(P,[S2,1, S2,2],[]) is true if R1 is true.

Level 2 consists of: rule(S2, 1, [H3,1, H3, 2], [P]) is true if R2,1 is true.

rule(S2, 2, [H3,3, H3, 4], [P]) is true if R2,2 is true.

Level 3 consists of: rule(H3,1, [],[P,S2,1]) is true if R3,1 is true.

rule(H3,2, [],[P,S2,1]) is true if R3,2 is true.

rule(H3,3, [],[P,S2,2]) is true if R3,13is true.

rule(H3,4, [],[P,S2,2]) is true if R3,4 is true.

In addition, each of the above 3-tuples has an associated I list of T and F statements so that further knowledge can be stored at a specific n-Cube node.

## 4.1 Elimination of Failure due to Inconsistency

The n-Cube schema allows the knowledge engineer to decompose a domain of expertise into small hierarchical partitions. This can assist in the simplification of the complex task of knowledge acquisition [Welbank83, Cosgrove91] and enables high quality explanations to be generated. For example, the n-Cube representation of the plant domain tutorial expert system, as shown in figure 5, hierarchically models this domain of expertise. "Deciduous" is defined by its parent classes "tree", "land based" and "plant domain"; to distinguish "deciduous" from "evergreen" a minimal rule set is used such that "has needles" identifies "evergreen" and "has green leaves" identifies "deciduous". As a result, inconsistencies can be identified by inspection of the rules in each P to H path.

## 4.2 Elimination of Duplication

Duplication seldom occurs in the n-Cube schema. The reasons for this are twofold. First, the minimum set of R associated with each P, S and H is used. Once P has been identified by this minimum rule set, then any Sub-Classes of S have only to be identified by a minimum set of rules that distinguish each Sub-Class of S. The second reason why the n-Cube schema is capable of eliminating duplication is because of its simple structure, therefore the knowledge engineer can rapidly identify duplicated rule antecedents.

## 4.3 Elimination of Incorrect Results

In the n-Cube representation of the plant domain, shown in figure 5, a new category known as "land based" is added to the knowledge-base. This new category is mutually exclusive with "seaweed", therefore both appear at the same level in the n-Cube's hierarchical representation. To ensure that only one of these categories succeeds in a knowledge-base search, the rules associated with these two classes are also mutually exclusive (rules Cube1 and Cube2). This allows specific incorrect results, that are inherent in standard rule-based systems, to be eliminated. Hence, these explicit hierarchical relationships simplify knowledge-base updating procedures and reduce the probability of conflict.

## 4.4 Enhanced Updating

The n-Cubes structured format allows the simple addition of new concepts and rules. For instance, if a new plant "mint" was to be added to the n-Cube's knowledge-base, shown in figure 6, then the knowledge-engineer knows that "mint" is a 'herb". This would result in the addition of a 3-tuple to the knowledge-base as follows :-

```
rule(mint,[nil],[plant domain,land based,herb])
                    is true if has small jagged green leaves
                    and has no flowers.
```

Consequently, the rules which focus the n-Cube search to the category "herb" remain constant, but the rules associated with the Sub-Classes of "herb" are investigated so that "mint", "lettuce" and "daffodil" can be uniquely identified.

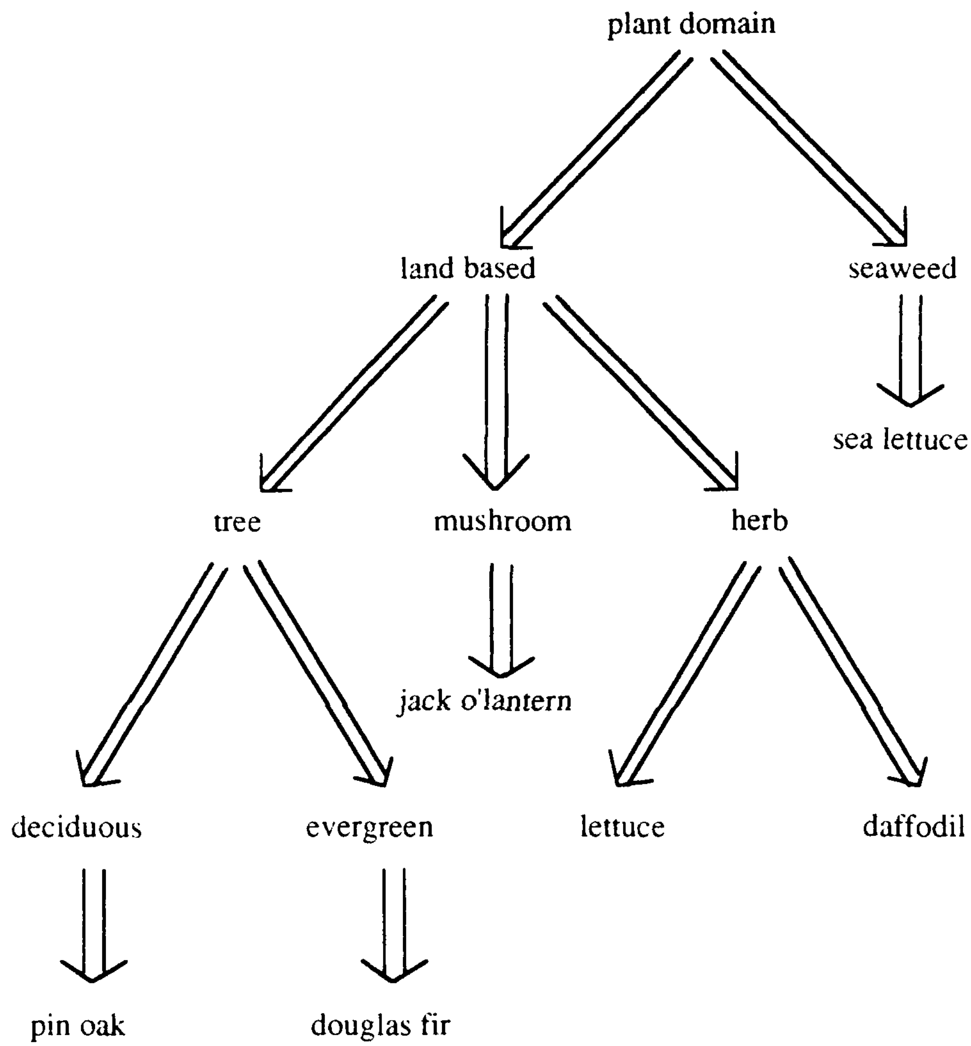


Figure 5. The n-Cube hierarchical description of the tutorial knowledge-base

Cube 1.
rule(land based,[tree,mushroom,herb],[plant domain]) is true if is land based.

Cube 2.
rule(seaweed,[sea lettuce],[plant domain]) is true if is not land based.

Cube 3.
rule(tree,[pin oak, douglas fir],[plant domain, land based]) is true if height > 5 feet and stem is woody.

Cube 4.
rule(mushroom,[jack o'lantern],[plant domain, land based]) is true if produces spores and is not green.

Cube 5.
rule(herb,[lettuce,daffodil],[plant domain, land based]) is true if stem is not woody.

Cube 6.
rule(sea lettuce,[nil],[plant domain,seaweed]) is true if has green leaves and has no flowers.

Cube 7.
rule(deciduous,[pin oak],[plant domain,land based, tree]) is true if has green leaves.

Cube 8.
rule(evergreen,[douglas fir],[plant domain,land based,tree]) is true if has needles.

Cube 9.
rule(lettuce,[nil],[plant domain,land based,herb]) is true if has green leaves and has no flowers.

Cube 10.
rule(daffodil,[nil],[plant domain,land based,herb]) is true if has green leaves and has yellow flowers.

Cube 11.
rule(pin oak,[nil],[plant domain,land based,tree,deciduous) is true *.

Cube 12.
rule(douglas fir,[nil],[plant domain,land based,tree,evergreen]) is true *.

* *Note Cube 11 and Cube 12 have no associated rules because they are the only Hypotheses that can be derived from their specific parents S.*

Figure 6. The Rules associated with the n-Cube Schema

# 5 CONCLUSION

Production-rule representation schemes are widely used in numerous expert system applications. The reasons for this are based upon evolution and the belief that such schemes are simple to understand and implement. Unfortunately, production-rule expert systems facilitate unstructured addition, deletion and modifications of rules. Consequently, such schemas are fraught with many undesirable features that include failure, duplication, incorrect results and knowledge-base updating complications. These features can be eliminated in the n-Cube schema by structured hierarchical decomposition and highly specific localised rule sets. Furthermore, the n-Cube can assist in knowledge acquisition by highlighting the deficiencies in a knowledge-base and may also enhance the quality of any derived explanations. As such, the n-Cube schema is a powerful tool that can be used to debug, develop and implement expert system applications.

# 6 ACKNOWLEDGEMENTS

Special thanks to both John Hickman and Philip Dart for their helpful suggestions and criticisms during this research.

# 7 REFERENCES

[Bundy83]
Bundy A. "The Computer Modelling of Mathematical Reasoning", Academic Press, 1983.

[Cosgrove90]
Cosgrove S.J., "Hierarchical knowledge classification and search mechanisms employed within the n-Cube expert system development tool", Libraries and Expert Systems Conference, Wagga Wagga, June 1990.

[Cosgrove91]
Cosgrove S.J., "Techniques for knowledge acquisition", DSTO Document, to be published 1991.

[Duda79]
Duda R., Gaschnig J., Hart P., "Model design in the PROSPECTOR consultant system for mineral exploration", in Expert Systems in the Microelectronic Age, D. Michie (ed.), Edinburgh, Edinburgh University Press, 1979.

[Findler79]
Findler N.V. "Associative networks representations and the use of knowledge by computers", Academic Press, 1979, pp 51-92.

[Luger89]
Luger G.F., Stubblefield W.A., "Artificial Intelligence and the Design of Expert Systems", Benjamin Cummings, 1990.

[McDermot82]
McDermot J., "R1: A rule-based configurer of computer systems", Artificial Intelligence, 1982, 19, pp 39-88.

[Minasi90a]
Minasi M., Asking the right questions, AI Expert, March 1990, pp 13-15.

[Minasi90b]
Minasi M., "What does an expert system look like ?", AI Expert, April 1990, pp 13-15.

[Minsky75]
Minsky M., "A Framework For Representing Knowledge", The Psychology of Computer Vision, P.H. Winston (ed.), New York: McGraw-Hill, 1975, pp 211-277.

[Neuron87]
Neuron Data, "Nexpert object fundamentals", Neuron data Inc., Palo Alto, California, 1988.

[Shortliffe76]
Shortliffe E.H., "Computer-based Medical Consultations: MYCIN, New York: Elsevier, 1976.

[Welbank83]
Welbank M. "A review of knowledge acquisition techniques for expert systems", British Telecom Research Laboratories, Martlesham Heath, Ipswich, England, 1983.

[Winston79]
Winston P.H. "Artificial Intelligence", Addison-Wesley, 1979.

[Winston84]
Winston P.H., Prendergast K.A. 'The AI Business", MIT Press, 1984.

# DISTRIBUTION

| | Copy No. |
|---|---|
| **Defence Science and Technology Organisation** | |
| Chief Defence Scientist ) | |
| Central Office Executive ) | 1 |
| Counsellor, Defence Science, London | Cnt Sht |
| Counsellor, Defence Science, Washington | Cnt Sht |
| Scientific Adviser, Defence Central | Cnt Sht |
| Scientific Adviser, Defence Intelligence Organisation | Cnt Sht |
| Scientific Adviser, Navy | Cnt Sht |
| Scientific Adviser, Air Force | Cnt Sht |
| Scientific Adviser, Army | Cnt Sht |
| **Electronics Research Laboratory** | |
| Director | 2 |
| Chief, Information Technology Division | 3 |
| Research Leader Information Processing and Fusion | 4 |
| Dr. G. Gibbon | 5 |
| Dr. S. Cosgrove (Author) | 6-9 |
| Publicity and Component Support Officer | 10 |
| Graphics and Documentation Support | 11 |
| **Libraries and Information Services** | |
| Director of Publications, Australian Government Publishing Service | 12 |
| Librarian, Technical Reports Centre, Defence Central Library Campbell Park | 13 |
| Manager, Document Exchange Centre, Defence Information Services | 14 |
| Defense Technical Information Center, United States | 15-16 |
| Defence Research Information Centre, United Kingdom | 17-18 |
| Director Scientific Information Services, Canada | 19 |
| Ministry of Defence, New Zealand | 20 |
| National Library of Australia | 21 |
| Defence Science and Technology Organisation Salisbury, Main Library | 22-23 |
| Librarian Defence Signals Directorate, Melbourne | 24 |
| British Library Document Supply Centre | 25 |
| **Spares** | |
| Defence Science and Technology Organisation Salisbury, Main Library | 26-31 |

Department of Defence

# DOCUMENT CONTROL DATA SHEET

| Page Classification | Privacy Marking/Caveat |
|---|---|
| UNCLASSIFIED | N/A |

| 1a. AR Number | 1b. Establishment Number | 2. Document Date | 3. Task Number |
|---|---|---|---|
| AR-006-420 | ERL-0520-RR | FEBRUARY 1991 | DST 90/151 |

| 4. Title | 5. Security Classification | 6. No. of Pages | 7. No. of Refs. |
|---|---|---|---|
| n REASONS WHY PRODUCTION-RULES ARE INSUFFICIENT MODELS FOR EXPERT SYSTEM KNOWLEDGE REPRESENTATION SCHEMES | U Document, U Title, U Abstract | 15 | 15 |

S (Secret) C (Confi ) R (Rest) U (Unclass)

* For UNCLASSIFIED docs with a secondary distribution LIMITATION, use (L)

| 8. Author(s) | 9. Downgrading/Delimiting Instructions |
|---|---|
| S.J. COSGROVE | N/A |

10a. Corporate Author and Address

Electronics Research Laboratory
PO Box 1600
SALISBURY SA 5108

10b. Task Sponsor

11. Officer/Position responsible for

Security........N/A........

Downgrading........N/A........

Approval for Release........D/ERL........

12. Secondary Release of this Document

APPROVED FOR PUBLIC RELEASE

Any enquiries outside stated limitations should be referred through DSTIC, Defence Information Services, Department of Defence, Anzac Park West, Canberra, ACT 2600.

13a. Deliberate Announcement

NO LIMITATION

13b. Casual Announcement (for citation in other documents)

- [√] No Limitation
- [ ] Ref. by Author & Doc No only

| 14. DEFTEST Descriptors | Classification | 15. DISCAT Subject Codes |
|---|---|---|
| Expert Systems | *Production Rules | 120901 |
| Knowledge Representation | *n-Cube | |

16. Abstract

The majority of expert system shells and fully developed expert system packages are founded upon simple observation and hypothesis production-rules. This paper investigates the use of such knowledge representation schemes by analysis of a simple, incomplete plant classification expert system. Furthermore, a new hierarchical knowledge representation scheme known as the n-Cube, which is suitable for classification applications, is discussed with reference to the disadvantages of rule-based systems.

| Page Classification |
|---|
| UNCLASSIFIED |
| Privacy Marking/Caveat |
| N/A |

| 16. Abstract (CONT.) | | |
|---|---|---|
| | | |
| 17. Imprint<br>Electronics Research Laboratory<br>PO Box 1600<br>SALISBURY SA 5108 | | |
| 18. Document Series and Number<br>ERL-0520-RR | 19. Cost Code<br>820441 | 20. Type of Report and Period Covered<br>ERL RESEARCH REPORT |
| 21 Computer Programs Used<br>N/A | | |
| 22. Establishment File Reference(s)<br>N/A | | |
| 23. Additional information (if required) | | |